Calibration Collapse: When Metacognitive Confidence Predictions Degenerate to Constants in Autonomous AI Systems

Vasyl Golubenko
TOV ZELTREX, Kyiv, Ukraine
Ukrainian Academy of Technology, Kyiv, Ukraine
vasyl.golubenko@zeltrex.com

April 2026

Abstract

Metacognitive architectures in autonomous AI agents promise self-aware decision-making through confidence-calibrated task selection. We report the first empirical documentation of *calibration collapse*—a failure mode in which a deployed metacognitive planning system degenerates to outputting a constant confidence value across all predictions, eliminating discriminative signal while preserving the structural appearance of self-awareness. In a production autonomous software development system operating over 14 days and 69 task executions, we observed the metacognitive planner consistently producing an identical confidence score of 0.7575 across all 60 recorded predictions, regardless of task category, model assignment, or historical quality outcomes. Root cause analysis revealed an empty calibration boundary table (0 rows despite 219 quality observations), indicating the system possessed the architectural scaffolding for calibration but never populated it—creating what we term a *metacognitive Potemkin village*. We formalize calibration collapse as a distinct failure class from the well-documented overconfidence bias in large language models, propose variance-based detection metrics, and outline a mandatory calibration bootstrapping protocol. Our findings carry implications for any AI system that relies on self-assessed confidence for autonomous decision-making.

Keywords: metacognition, confidence calibration, autonomous agents, failure modes, self-improving systems, large language models

1 Introduction

The deployment of autonomous AI agents in software engineering and knowledge work has accelerated rapidly, with systems now capable of independently selecting tasks, generating code, and evaluating their own outputs [Fang et al., 2025, He et al., 2025]. A critical architectural component of such systems is *metacognitive planning*—the ability to assess one's own capabilities and allocate effort accordingly [Wang et al., 2025, Shinn et al., 2023]. When functioning correctly, metacognitive confidence scores enable optimal task selection: high-confidence tasks proceed efficiently, while low-confidence tasks receive additional resources or are deferred.

Recent work has established that large language models (LLMs) exhibit systematic overconfidence in their predictions [Steyvers et al., 2025, Ghosh and Panday, 2026, Xiong et al., 2024]. The DMC framework demonstrated that decoupling metacognition from cognition improves self-assessment accuracy [Wang et al., 2025], while studies on reward model calibration revealed inherent biases toward high confidence scores regardless of output quality [Leng et al., 2025]. However, these studies focus on a spectrum of miscalibration—the confidence predictions may be biased but retain *some* discriminative signal.

We report a qualitatively different failure mode: *calibration collapse*, in which a metacognitive system's predictions degenerate to a single constant value, losing all discriminative capacity while maintaining the structural appearance of functioning self-assessment. This is not overconfidence (predicting too-high values) or underconfidence (predicting too-low values), but rather the complete absence of prediction—the system outputs the same number for every input.

Our contribution is threefold: (1) we formally define calibration collapse and distinguish it from existing miscalibration categories; (2) we provide empirical evidence from a production autonomous development system, documenting the phenomenon across 60 predictions over 14 days; (3) we propose detection metrics, prevention mechanisms, and a mandatory bootstrapping protocol for deployed metacognitive systems.

2 Related Work

2.1 LLM Confidence and Calibration

Research on LLM confidence estimation has grown substantially. Xiong et al. [Xiong et al., 2024] provided a systematic evaluation of confidence elicitation methods, finding LLMs to be consistently overconfident across multiple strategies. Geng et al. [Geng et al., 2024] surveyed confidence estimation and calibration techniques, establishing evaluation frameworks based on Expected Calibration Error (ECE) and Brier scores. Steyvers et al. [Steyvers et al., 2025] demonstrated a significant gap between what LLMs actually know and what users perceive them to know. Leng et al. [Leng et al., 2025] specifically addressed overconfidence arising from RLHF training, showing that reward models exhibit inherent bias toward high-confidence outputs regardless of actual quality.

Ghosh and Panday [Ghosh and Panday, 2026] documented the Dunning-Kruger effect in LLMs—a pattern where models are most overconfident on tasks they perform worst at. This connects to our finding: when confidence collapses to a constant, the system exhibits neither the Dunning-Kruger pattern nor appropriate calibration—it exhibits *no pattern at all*.

2.2 Metacognition in AI Systems

Wang et al. [Wang et al., 2025] proposed the DMC framework for decoupling metacognition from cognition in LLMs, achieving improved self-assessment through architectural separation. Becker et al. [Becker et al., 2025] argued that truly self-improving agents require intrinsic metacognitive learning across three dimensions: knowledge, planning, and evaluation. Yildirim et al. [Yildirim et al., 2025] found evidence for limited metacognitive monitoring abilities in LLMs at the level of internal activations, while Mondorf and Plank [Mondorf and Plank, 2025] presented counterevidence suggesting these abilities are qualitatively limited.

Our work extends this literature by documenting what happens when metacognitive architecture is present but non-functional—a case where the system has been designed for self-assessment but the calibration mechanism fails silently.

2.3 Autonomous Agent Failure Modes

Cemri et al. [Cemri et al., 2025] analyzed 1,642 execution traces across seven multi-agent systems, finding failure rates of 41–87% and proposing a failure taxonomy. Roig [Roig, 2025] provided qualitative analysis of LLM failure patterns in agentic scenarios. Our work contributes a new failure class—calibration collapse—to this taxonomy.

3 System Description

3.1 Night Shift Architecture

Night Shift is an autonomous software development agent deployed on a dedicated server (Hetzner GEX44, RTX 4000 SFF Ada GPU). It operates on an hourly dispatch cycle, selecting tasks from a backlog of software engineering work, executing them using LLM inference, and evaluating the results. The system comprises approximately 46,000 lines of Python across five subsystems: Pulse (quota governance), Mind (task management and metacognition), Hands (execution and integration), Reflect (quality assessment and learning), and Evolution (genetic algorithm for prompt optimization).

3.2 Metacognitive Planning Module

The metacognitive planner (328 LOC) computes a confidence score for task selection using a four-axis weighted composite:

$$C = w_{\text{weak}} \cdot S_{\text{weak}} + w_{\text{roi}} \cdot S_{\text{roi}} + w_{\text{avoid}} \cdot S_{\text{avoid}} + w_{\text{bloom}} \cdot S_{\text{bloom}} \tag{1}$$

where S_{weak} (weight 0.30) scores based on category quality history; S_{roi} (weight 0.30) estimates return-on-investment; S_{avoid} (weight 0.20) penalizes repeatedly skipped tasks; and S_{bloom} (weight 0.20) matches cognitive complexity via Bloom's Taxonomy. The final confidence is clamped to $[0, 1]$.

3.3 Calibration Boundary System

A boundary classification module (250 LOC) classifies each (category, output_type) pair into three capability zones: CAPABLE (mean quality ≥ 7.0 with ≥ 5 samples), ZPD (mean quality $\in [4.0, 7.0)$), and BEYOND (mean quality < 4.0). These classifications are stored in the `mc_boundaries` table and intended to feed into the weakness axis.

3.4 Quality Assessment

Quality scores (1–10 scale) are produced by a multi-stage assessor (1,221 LOC) combining heuristic analysis with LLM-as-Judge scoring in a 25/75 blend. Inflation is capped at +2 above heuristic baseline; truncation incurs −2 penalty.

4 Methodology

We analyzed data from 14 days of autonomous operation (March 19–April 3, 2026), during which the system executed 69 tasks across 7 categories using 3 model families (local Ollama qwen2.5-coder:14b, Claude Sonnet 4.5, Claude Haiku 4.5). Data was extracted from three SQLite databases: `mc_store.db` (60 confidence predictions), `zpd_tracker.db` (219 quality observations), and `monitor_trends.db` (14 daily snapshots).

We evaluated calibration using prediction variance $\text{Var}(C)$, confidence entropy $H(C) = -\sum_k p_k \log_2 p_k$, and Expected Calibration Error:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{n} |\text{acc}(B_m) - \text{conf}(B_m)| \tag{2}$$

5 Results

5.1 Confidence Distribution

All 60 confidence predictions were identical: $C = 0.7575$ (Table 1). The prediction variance is $\mathrm{Var}(C) = 0.0$ and the confidence entropy is $H(C) = 0.0$ bits, indicating complete calibration collapse.

Table 1: Confidence predictions by model (14-day window).

Model	Tasks	Mean Confidence	Std Dev
qwen2.5-coder:14b	39	0.7575	0.0
claude-sonnet-4-5	13	0.7575	0.0
claude-haiku-4-5	8	0.7575	0.0
All models	**60**	**0.7575**	**0.0**

5.2 Actual Quality Distribution

In contrast, actual quality scores exhibited substantial variance (Table 2), demonstrating that meaningful signal existed but was not captured by the confidence predictor.

Table 2: Actual quality scores by category (14-day window, $N = 69$).

Category	Tasks	Mean Q	Std Dev	Min	Max
META	19	6.20	0.74	5.0	7.2
HUB	18	5.49	0.47	5.0	7.0
BRIDGE	13	5.98	0.62	5.0	7.2
NEXUS	8	5.58	0.78	4.5	6.5
RESEARCH	7	6.11	0.98	4.5	7.0
LIVINGCORP	2	6.50	0.71	6.0	7.0
OPTIMIZATION	2	6.00	0.00	6.0	6.0
Overall	**69**	**5.79**	**0.72**	**4.5**	**7.2**

The ECE for the collapsed predictor is $|0.532 - 0.7575| = 0.226$, indicating substantial overconfidence.

5.3 Root Cause: Empty Calibration Table

The `mc_boundaries` table contained exactly 0 rows despite 219 quality observations being available in the ZPD tracker database. Without boundary data, the weakness axis defaulted to a uniform score across all categories, collapsing the primary discriminative signal (30% of total weight).

5.4 Downstream Impact

The calibration collapse produced measurable effects: (1) the planner selected the same task (`ingest-300`) in 452 out of 507 planning iterations (89.2% repetition); (2) despite 507 planning cycles, only 69 tasks executed; (3) system logs showed structured MC rationale that masked the absence of actual decision-making.

5.5 ZPD Band Analysis

The ZPD tracker independently classified tasks into developmental bands (Table 3), providing ground truth for what a functioning calibration system should capture.

Table 3: ZPD band distribution (14-day window).

Band	Tasks	Mean Quality	Expected Confidence
beyond_reach	0	—	Low (< 0.3)
zpd	51	5.40	Medium (0.4–0.7)
actual_development	18	7.29	High (> 0.7)

A calibrated system should assign higher confidence to "actual_development" tasks (quality 7.29) and lower to "zpd" tasks (quality 5.40). Instead, both received identical 0.7575.

6 Discussion

6.1 Calibration Collapse as a Distinct Failure Mode

We propose a taxonomy of metacognitive failures:

1. **Overconfidence** [Steyvers et al., 2025, Ghosh and Panday, 2026]: $E[C] > E[\text{acc}]$ but $\text{Var}(C) > 0$.
2. **Underconfidence**: $E[C] < E[\text{acc}]$ but $\text{Var}(C) > 0$.
3. **Dunning-Kruger** [Ghosh and Panday, 2026]: Inverse confidence-accuracy relationship. Signal exists but is inverted.
4. **Calibration collapse** (this work): $\text{Var}(C) = 0$. No discriminative signal.

Calibration collapse is uniquely dangerous because it preserves the *appearance* of functioning metacognition while providing zero actual self-assessment.

6.2 The Metacognitive Potemkin Village

We introduce the concept of a *metacognitive Potemkin village*: an AI system that possesses complete architectural scaffolding for self-assessment but in which the calibration feedback loop is broken. In our system, the architecture included a 4-axis scorer, capability boundary classifier, knowledge graph (972 nodes, 2,998 edges), and ZPD tracker (219 observations)—yet a single missing link (population of `mc_boundaries`) rendered the entire apparatus non-functional.

6.3 Detection Framework

We propose three detection metrics:

Metric 1: Confidence Variance Monitor. Track rolling variance $V_w = \text{Var}(C_{t-w}, \ldots, C_t)$; alert when $V_w < \epsilon$ for $w \geq 20$.

Metric 2: Confidence-Quality Correlation. Compute $r_{CQ} = \text{Corr}(C, Q)$; alert when $|r_{CQ}| < 0.1$ over $n \geq 30$.

Metric 3: Unique Confidence Count. Count distinct values U_w in window; alert when $U_w/w < 0.1$.

6.4 Prevention: Mandatory Calibration Bootstrapping

We propose: (1) pre-deployment validation requiring ≥ 50 calibration observations; (2) warm-up phase with random selection; (3) continuous liveness checking for ≥ 3 distinct values per 20-observation window; (4) automatic fallback to priority-based selection on collapse detection.

6.5 Limitations

Our study is based on a single production system. The constant 0.7575 may be specific to the weighting configuration. We did not conduct ablation studies. Quality scores are partially automated.

7 Conclusion

We documented and formalized *calibration collapse*—a previously unreported failure mode in which a metacognitive AI system's confidence predictions degenerate to a constant. In a production system, this manifested as $C = 0.7575$ across all 60 predictions over 14 days, despite actual quality varying from 4.5 to 7.2.

The root cause—an empty calibration boundary table despite abundant quality data—illustrates the fragility of metacognitive architectures. We propose variance-based detection metrics and a mandatory bootstrapping protocol. As autonomous AI agents are deployed in consequential domains, continuous calibration monitoring becomes a safety requirement.

References

J. Fang, Y. Peng, X. Zhang, Y. Wang, X. Yi, G. Zhang, Y. Xu, B. Wu, S. Liu, Z. Li, Z. Ren, N. Aletras, X. Wang, H. Zhou, and Z. Meng. A comprehensive survey of self-evolving AI agents: A new paradigm bridging foundation models and lifelong agentic systems. *arXiv preprint arXiv:2508.07407*, 2025.

J. He, C. Treude, and D. Lo. LLM-based multi-agent systems for software engineering: Literature review, vision, and the road ahead. *ACM Transactions on Software Engineering and Methodology*, 34(5):124:1–124:30, 2025. 10.1145/3712003.

G. Wang, W. Wu, G. Ye, Z. Cheng, X. Chen, and H. Zheng. Decoupling metacognition from cognition: A framework for quantifying metacognitive ability in LLMs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25353–25361, 2025. 10.1609/aaai.v39i24.34723.

N. Shinn, F. Cassano, E. Berman, A. Gopinath, K. Narasimhan, and S. Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

M. Steyvers, H. Tejeda, A. Kumar, C. Belem, S. Karny, X. Hu, L. W. Mayer, and P. Smyth. What large language models know and what people think they know. *Nature Machine Intelligence*, 7:221–231, 2025. 10.1038/s42256-024-00976-7.

S. Ghosh and M. Panday. The Dunning-Kruger effect in large language models: An empirical study of confidence calibration. *arXiv preprint arXiv:2603.09985*, 2026.

M. Xiong, Z. Hu, X. Lu, Y. Li, J. Fu, J. He, and B. Hooi. Can LLMs express their uncertainty? An empirical evaluation of confidence elicitation in LLMs. In *Proceedings of ICLR*, 2024.

J. Leng, C. Huang, B. Zhu, and J. Huang. Taming overconfidence in LLMs: Reward calibration in RLHF. In *Proceedings of ICLR*, 2025.

J. Geng, F. Cai, Y. Wang, H. Koeppl, P. Nakov, and I. Gurevych. A survey of confidence estimation and calibration in large language models. In *Proceedings of NAACL*, pages 6577–6595, 2024.

T. Becker et al. Truly self-improving agents require intrinsic metacognitive learning. *ICML 2025 Position Paper*, 2025.

I. Yildirim et al. Language models are capable of metacognitive monitoring and control of their internal activations. *arXiv preprint arXiv:2505.13763*, 2025.

M. Mondorf and B. Plank. Evidence for limited metacognition in LLMs. *arXiv preprint arXiv:2509.21545*, 2025.

M. Cemri, M. Z. Pan, S. Yang, L. A. Agrawal, B. Chopra, R. Tiwari, K. Keutzer, A. Parameswaran, D. Klein, K. Ramchandran, M. Zaharia, J. E. Gonzalez, and I. Stoica. Why do multi-agent LLM systems fail? *arXiv preprint arXiv:2503.13657*, 2025.

J. V. Roig. How do LLMs fail in agentic scenarios? A qualitative analysis. *arXiv preprint arXiv:2512.07497*, 2025.

L. Sun, Y. Yang, Q. Duan, Y. Shi, C. Lyu, Y.-C. Chang, C.-T. Lin, and Y. Shen. Multi-agent coordination across diverse applications: A survey. *arXiv preprint arXiv:2502.14743*, 2025.